Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1 – 35. Cancelled

36. (Currently Amended) ~~The method as defined in claim 34,~~ A method executed on a computer system for providing themes for controls of a first application and a second application in a graphical operating system environment, method comprising:

receiving a first render request for a first control of the first application, the first control capable of being rendered according to properties in a non-binary theme file;

receiving a second render request for a second control of the second application, the second control incapable of being rendered according to properties in a non-binary theme file;

converting the non-binary theme file to a shared binary theme file;

in response to the first render request, accessing the shared binary theme file to retrieve theme property data for the first control, wherein the act of accessing the shared binary theme file comprises:

retrieving an offset into a class data section of the shared binary theme file, the class data section having theme property data for a class in binary format;

performing a binary search for class property data at the offset;

determining if class property data exists at the offset;

jumping to a global data section of the shared binary theme file having global theme property data, if no class property data is found; and

retrieving global theme property data from the global data section;

in response to the second render request, accessing the shared binary theme file to retrieve theme property data for the second control; and

rendering the first and second control according to the retrieved theme property data.

37. (Currently Amended) ~~The method as defined in claim 34,~~ A method executed on a computer system for providing themes for controls of a first application and a second application in a graphical operating system environment, method comprising:

receiving a first render request for a first control of the first application, the first control capable of being rendered according to properties in a non-binary theme file;

receiving a second render request for a second control of the second application, the second control incapable of being rendered according to properties in a non-binary theme file;

converting the non-binary theme file to a shared binary theme file;

in response to the first render request, accessing the shared binary theme file to retrieve theme property data for the first control, wherein the act of accessing the shared binary theme file comprises:

retrieving an offset into a part jump table section of the shared binary theme file, the part jump table section having theme property data for a part in binary format;

performing a binary search for part property data at the offset;

determining if part property data exists at the offset;

jumping to a class data section of the shared binary theme file having theme property data for a class, if no part property data is found; and

retrieving class theme property data from the class data section;

in response to the second render request, accessing the shared binary theme file to retrieve theme property data for the second control; and

rendering the first and second control according to the retrieved theme property data.

38. (Currently Amended) ~~The method as defined in claim 34,~~ A method executed on a computer system for providing themes for controls of a first application and a second application in a graphical operating system environment, method comprising:

receiving a first render request for a first control of the first application, the first control capable of being rendered according to properties in a non-binary theme file;

receiving a second render request for a second control of the second application, the second control incapable of being rendered according to properties in a non-binary theme file;

converting the non-binary theme file to a shared binary theme file;

in response to the first render request, accessing the shared binary theme file to retrieve theme property data for the first control, wherein the step of accessing the shared binary theme file comprises:

retrieving a memory offset into a part jump table section of the shared binary theme file;

retrieving from the part jump table section a second memory offset into a state jump table section;

jumping to the second memory offset of the shared binary theme file having state theme property data; and

retrieving state theme property data from the state theme property data section;

in response to the second render request, accessing the shared binary theme file to retrieve theme property data for the second control; and

rendering the first and second control according to the retrieved theme property data.

39. (New) The method as defined in claim 36, wherein the first and second render requests include a theme handle and a component state.

40. (New) The method as defined in claim 36, further comprising:

creating a visual style for a set of controls of the first application and the second application, wherein creating comprises:

selecting controls, from a schema file of controls, that are desired to have a defined visual style, each control being defined by a unique class name;

assigning properties to the selected controls according to the defined visual style so that each selected control has assigned properties;

grouping the pairs of selected-controls and corresponding assigned properties for the defined visual style together in the non-binary theme file.

41. (New) The method as defined in claim 40, further comprising loading the shared binary theme file into a shared memory so that a visual style can be used to render controls.

42. (New) The method as defined in claim 40, wherein the controls defined within the schema file of controls have one or more part names associated with at least one class name, and the converting act further comprises creating a part property data section in the shared binary theme file, the part property data section having the one or more part names and the assigned properties.

43. (New) The method as defined in claim 40, wherein the controls defined within the schema file of controls have one or more state names associated with at least one defined part name, and the converting act further comprises creating a state property data section in the shared binary theme file, the state property data section having the one or more state names and the assigned properties.

44. (New) The method as defined in claim 40, further comprising:

identifying some properties as global theme property data; and

creating in the shared binary theme file a global theme data section having the global theme property data to be used when a class name, part name, or state name cannot be found in the shared binary theme file.

45. (New) The method as defined in claim 46, wherein a list of available properties is within the schema file of controls, that may be selected in the selecting step for each control, part and state.

46. (New) The method as defined in claim 37, wherein the first and second render requests include a theme handle and a component state.

47. (New) The method as defined in claim 37, further comprising:

creating a visual style for a set of controls of the first application and the second application, wherein creating comprises:

selecting controls, from a schema file of controls, that are desired to have a defined visual style, each control being defined by a unique class name;

assigning properties to the selected controls according to the defined visual style so that each selected control has assigned properties;

grouping the pairs of selected-controls and corresponding assigned properties for the defined visual style together in the non-binary theme file.

48. (New) The method as defined in claim 47, further comprising:

identifying some properties as global theme property data; and

creating in the shared binary theme file a global theme data section having the global theme property data to be used when a class name, part name, or state name cannot be found in the shared binary theme file.

49. (New) The method as defined in claim 48, wherein a list of available properties is within the schema file of controls, that may be selected in the selecting step for each control, part and state.

50. (New) The method as defined in claim 49, wherein the act of converting comprises:

identifying a derived property for a control;

associating a unique numeric identifier with the derived property to create a derived property identifier;

identifying one or more primitive properties for each derived property, wherein each primitive property has associated property data having a length;

associating a unique numeric identifier with each primitive property, to create a primitive property identifier;

calculating the lengths of each of the associated property data;

selecting a derived property identifier;

writing a binary tagged data module to a tagged data memory offset in the class data section of the shared binary file wherein the binary tagged data module contains the selected derived property identifier, the one or more primitive property identifiers, the associated property values, and each of the property values' lengths; and

writing an associated parent part offset after each binary tagged data module, the associated parent part offset being a memory offset into the global class section.

51. (New) The method as defined in claim 38, wherein the first and second render requests include a theme handle and a component state.

52. (New) The method as defined in claim 38, further comprising:

creating a visual style for a set of controls of the first application and the second application, wherein creating comprises:

selecting controls, from a schema file of controls, that are desired to have a defined visual style, each control being defined by a unique class name;

assigning properties to the selected controls according to the defined visual style so that each selected control has assigned properties;

grouping the pairs of selected-controls and corresponding assigned properties for the defined visual style together in the non-binary theme file.

53. (New) The method as defined in claim 52, further comprising:

identifying some properties as global theme property data; and

creating in the shared binary theme file a global theme data section having the global theme property data to be used when a class name, part name, or state name cannot be found in the shared binary theme file.

54. (New) The method as defined in claim 53, wherein a list of available properties is within the schema file of controls, that may be selected in the selecting step for each control, part and state.

55. (New) The method as defined in claim 54, wherein the act of converting comprises:

identifying a derived property for a control;

associating a unique numeric identifier with the derived property to create a derived property identifier;

identifying one or more primitive properties for each derived property, wherein each primitive property has associated property data having a length;

associating a unique numeric identifier with each primitive property, to create a primitive property identifier;

calculating the lengths of each of the associated property data;

selecting a derived property identifier;

writing a binary tagged data module to a tagged data memory offset in the class data section of the shared binary file wherein the binary tagged data module contains the selected derived property identifier, the one or more primitive property identifiers, the associated property values, and each of the property values' lengths; and

writing an associated parent part offset after each binary tagged data module, the associated parent part offset being a memory offset into the global class section.

56. (New) The method as defined in claim 55, wherein the act of converting further comprises:

obtaining the memory offset of a binary tagged data module for a state; and

writing the memory offset to a second memory offset in a state jump table in the shared binary theme file.

57. (New) The method as defined in claim 56, wherein the act of converting further comprises:

writing the second memory offset to a third memory offset in a part jump table in the shared binary theme file.